

Dokumentation der XML Schnittstellen

Sparzähler mit WiNG Architektur

Inhaltsverzeichnis

1	Schnittstelle zum Abfragen von ComModul-Daten	2
1.1	Abfrage von Lastprofilwerten (Veraltet – Deprecated)	2
1.1.1	Format des GET-Requests	2
1.1.2	Ablauf der Kommunikation	3
1.1.3	Format der Antwort	3
1.1.4	Formale Formatdefinition	5
1.1.5	Fehlerhandling	7
1.2	Abfrage der ComModul-Uhrzeit (Veraltet – Deprecated)	8
1.2.1	Format des GET-Requests	8
1.2.2	Format der Antwort	8
1.2.3	Formale Formatdefinition	9
1.2.4	Fehlerhandling	11
1.3	Abfrage von Identifikationsmerkmalen und Systeminformationen (Veraltet – Deprecated)	12
1.3.1	Format des GET-Requests	12
1.3.2	Format der Antwort	12
1.3.3	Formale Formatdefinition	15
1.3.4	Fehlerhandling	18

1 Schnittstelle zum Abfragen von ComModul-Daten

Die XML-Schnittstelle zum Abfragen von ComModul-Daten definiert HTTP-Anfragen an den Web-Server des ComModul und deren Antworten. Als Trägerformat für die Antwort des Web-Servers wird XML eingesetzt.

Folgende Abfragen sind definiert:

1. Abfrage von Lastprofilwerten
2. Abfrage der ComModul-Uhrzeit
3. Abfrage von Identifikationsmerkmalen und Systeminformationen

Die XML-Schnittstellen sind ausschließlich zur Wahrung der Kompatibilität zu bestimmten Versionen von Yellometer und Stromradar noch vorhanden. Software soll generell die in WiNG neu implementierten UPnP Schnittstellen verwenden.

1.1 Abfrage von Lastprofilwerten (Veraltet – Deprecated)

Die Abfrage von Lastprofilwerten erfolgt über einen HTTP-GET-Aufruf der Seite getPowerProfile.html und zugehörigen Parametern.

1.1.1 Format des GET-Requests

Die URL der Abfrage lautet:

`http://<IP des ComModul>/InstantView/request/getPowerProfile.html`

Mit den folgenden Parametern kann die Abfrage der im ComModul vorgehaltenen Daten eingegrenzt werden:

ts	Startzeitpunkt der Abfrage mit folgenden Möglichkeiten: a) im Standarddatumsformat, z. B. "130628130245" für den 28.06.2013, 13:02:45 Uhr b) "0", wenn die letzten n Werte gelesen werden sollen gemäß nachfolgend beschriebenen Parameter n Dieser Parameter muss immer angegeben werden. Es wird kein Defaultwert angenommen.
N	Maximale Anzahl der zurück gelieferten Werte (Default ist „100“). Die maximale Anzahl ist durch die Größe des ComModul-Ringpuffers nach oben auf 86400 Werte beschränkt, d.h. für 1-Sekundenwerte aus 24 Stunden. Eine weitere Einschränkung ist die Größe des internen (software-seitigen) Kommunikations-Buffer von 64KB. Die Berechnungsformel zur Bestimmung der maximal zulässigen Werte-Anzahl lautet wie folgt: Max. Anzahl = (Komm.-Buffergröße – Header) / (Anzahl Stellen Leistungswert + 1) = (65536 – 46) / (5 + 1) = 10.919 Werte
param	Bestimmt, welcher Parameter ausgelesen werden soll (
resformat	Format der Antwort (Veraltet – Deprecated)).
format	wie resformat (Veraltet – Deprecated)

Beispiel 1

Abfrage von 10 Wirkleistungs-Werten ab dem 04.09.2013 um 11 Uhr 47 und 0 Sekunden im Format „1“:

```
http://<IP des ComModul>/InstantView/request/getPowerProfile.html?  
ts=130904114700&n=10&format=1
```

Beispiel 2

Abfrage des „aktuellen“ Messwerts zum Zeitpunkt der Verarbeitung im ComModul, dies entspricht in etwa der Anzeige der aktuellen Abnahme über die Startseite im Browser.

```
http://<IP des ComModul>/InstantView/request/getPowerProfile.html?n=1
```

1.1.2 Ablauf der Kommunikation

Bei einer kontinuierlichen Abfrage von Lastprofil-Werten bietet es sich an, für den ersten Request, zunächst die letzten n Werte (hier z. B. 100 Werte) aus dem ComModul auszulesen. Die Abfrage hierzu lautet wie folgt:

```
http://<IP des ComModul>/InstantView/request/getPowerProfile.html?  
ts=0&n=100&format=1
```

Anschließend können neue Werte ausgelesen werden, indem als Startzeitpunkt der neuen Abfrage der Endzeitpunkt der letzten Anfrage verwendet wird. Um Überschneidungen zu vermeiden (Werte des Endzeitpunkts bzw. Startzeitpunkts werden mit beiden Anfragen geliefert), kann als Startzeitpunkt auch der letzte Endzeitpunkt inkrementiert um die Sample-Rate verwendet werden.

War zum Beispiel dem letzten Wert aus der vorhergehenden Abfrage die Uhrzeit „130628130244“, also 13:02 Uhr und 44 Sekunden (am 28.06.2013) zugeordnet, kann die darauf folgende Wertefolge bei einer Sample-Rate von 1 Sekunde wie folgt abgefragt werden:

```
http://<IP des ComModul>/InstantView/request/getPowerProfile.html?  
ts=130628130245&n=100&format=1
```

Zu beachten ist hier, dass auch weniger als die angegebene Menge an Werten zurückgeliefert werden kann, abhängig davon, wie viele Messwerte ab dem angegebenen Zeitpunkt tatsächlich aufgezeichnet wurden bzw. aufgezeichnet werden konnten.

Konnten während eines Zeitraums einzelne Werte nicht aus dem Zähler ausgelesen werden, wird statt eines Zahlenwerts jeweils das Kürzel "NAN" ausgegeben. Eine Lücke in der Aufzeichnung und Ausgabe entsteht deswegen nicht.

1.1.3 Format der Antwort

Das Format der Antwort sieht wie folgt aus (Angaben in eckigen Klammern stehen stellvertretend für den Typ bzw. Inhalt der dort zu erwartenden Angabe):

```
<?xml version="1.0" encoding="utf-8" ?>  
<values xmlns="/InstantView/request/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-  
instance" xsi:schemaLocation="/InstantView/request/ file:powerProfile.xsd">  
  <header>
```

```

<name>[Angeforderter Parameter]</name>
<obis>[OBIS-Kennung]</obis>
<startts>[Startzeitpunkt im Format YYMMTTHHMMSS]</startts>
<endts>[Endzeitpunkt im Format YYMMTTHHMMSS]</endts>
<samplerate>[Sample-Rate, bisher immer 1]</samplerate>
<no>[Anzahl der zurück gelieferten Werte]</no>
<error>[true/false]</error>
</header>
<v>[Ergebnis-Startwert]</v>
<v>[1. Ergebnis-Wert nach dem Startwert]</v>
<v>[2. Ergebnis-Wert nach dem Startwert]</v>
<v>...</v>
<v>[Letzter Ergebniswert]</v>
</values>

```

Die Elemente der XML-Tags haben hierbei die folgende Bedeutung:

name	Name des Parameters (identisch zur Anfrage).
obis	OBIS-Kennung des Parameters gemäß IEC/DIN-EN
startts	Startzeitpunkt im Standarddatumsformat, Zeitstempel des ersten Wertes im Format YYMMDDhhmmss.
endts	Endzeitpunkt im Standarddatumsformat, Zeitstempel des letzten Wertes im Format YYMMDDhhmmss.
samplerate	Aufzeichnungsrate der Wert in Sekunden (zur Bestimmung der Einzelzeitstempel aus dem Startzeitpunkt und der Position des Einzelwertes). Aktuell ist Sample-Rate fest 1 Sekunde.
no	Anzahl der zurück gelieferten Werte.
error	Falls ein Fehler aufgetreten ist „true“ - ansonsten „false“
v	Ergebniswert. Aktuell ist die Einheit des Ergebniswerts immer fest „Watt“, und es werden nur Vorkommastellen geliefert.

Die einzelnen Wert werden innerhalb der <v>-Elemente zurückgeliefert, beginnend beim Startzeitpunkt und sich jeweils um die Samplerate erhöhend bis zum Endzeitpunkt. Die Anzahl der Werte im Feld <no> muss dabei nicht der ursprünglich geforderten Anzahl entsprechen, falls diese (noch) nicht geliefert werden kann. Die Messwerte liegen jedoch immer in der angegebenen Reihenfolge entsprechend dem Abstand der einfachen Sample-Rate und ohne Lücke vor.

Beispiel:

Es werden 6 Wirkleistungswerte ab dem 28.06.2013, 13:02 und 40 Sekunden geliefert. Um 13:02:40 betrug die Wirkleistung 101 Watt, um 13:02:41 betrug die Wirkleistung 103 Watt usw.

```

<?xml version="1.0" encoding="utf-8" ?>
<values xmlns="/InstantView/request/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="/InstantView/request/ file:powerProfile.xsd">
  <header>
    <name>Wirkleistung</name>
    <obis>OBIS 1-1:16.7.0</obis>
    <startts>130628130240</startts>
    <endts>130628130245</endts>
    <samplerate>1</samplerate>
    <no>6</no>
    <error>false</error>
  </header>
  <v>101</v>
  <v>103</v>
  <v>107</v>

```

```
<v>120</v>
<v>122</v>
<v>119</v>
</values>
```

1.1.4 Formale Formatdefinition

Die formale Formatdefinition ist in der Datei /InstantView/request/powerProfile.xsd hinterlegt:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="/InstantView/request/"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:cm="/InstantView/request/" element-
FormDefault="qualified" attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation>
      Antwort des ComModuls auf getPowerProfile-Requests
    </xs:documentation>
  </xs:annotation>
  <xs:element name="values" type="cm:t_values"></xs:element>

  <xs:complexType name="t_values">
    <xs:sequence>
      <xs:element name="header" type="cm:t_header" maxOccurs="1" minOccurs="1">
        </xs:element>
      <xs:choice>
        <xs:element name="v" maxOccurs="unbounded" minOccurs="0">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:pattern value="[0-9]+(\.[0-9]+)?"></xs:pattern>
              <xs:pattern value="NAN"></xs:pattern>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
        <xs:element name="error" type="cm:t_error" maxOccurs="1" minOccurs="0">
          </xs:element>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="t_header">
    <xs:sequence>
      <xs:element name="name" maxOccurs="1" minOccurs="1">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Wirkleistung"></xs:enumeration>
            <xs:enumeration value="ERR"></xs:enumeration>
            <xs:enumeration value=""></xs:enumeration>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="obis" maxOccurs="1" minOccurs="1">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="OBIS 1-1:16.7.0"></xs:enumeration>
            <xs:enumeration value="ERR"></xs:enumeration>
            <xs:enumeration value=""></xs:enumeration>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="startts" maxOccurs="1" minOccurs="1">
        <xs:simpleType value="([0-9]{12}[nws]?){0,1}">
          <xs:restriction base="xs:string">
            <xs:pattern value="([0-9]{12}[nws]?){0,1}"></xs:pattern>
            <xs:pattern value="ERR"></xs:pattern>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="endts" maxOccurs="1" minOccurs="1">
        <xs:simpleType value="([0-9]{12}[nws]?){0,1}">
          <xs:restriction base="xs:string">
```

Formatiert: Englisch (USA)

```

        <xs:pattern value="{[0-9]{12}[nws]?}"{0,1}"></xs:pattern>
        <xs:pattern value="ERR"></xs:pattern>
    </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="samplerate" maxOccurs="1" minOccurs="1">
    <xs:simpleType value="([1-9]{1}[0-9]*){0,1}">
        <xs:restriction base="xs:positiveInteger">
            <xs:pattern value="([1-9]{1}[0-9]*){0,1}"></xs:pattern>
            <xs:pattern value="ERR"></xs:pattern>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="no" maxOccurs="1" minOccurs="1">
    <xs:simpleType value="([0-9]+){0,1}">
        <xs:restriction base="xs:positiveInteger">
            <xs:pattern value="([0-9]+){0,1}"></xs:pattern>
            <xs:pattern value="ERR"></xs:pattern>
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<xs:element name="error" type="xs:boolean" maxOccurs="1" minOccurs="1"></xs:element>
</xs:sequence>
</xs:complexType>

<xs:complexType name="t_error">
    <xs:sequence>
        <xs:element name="id" maxOccurs="1" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:positiveInteger">
                    <xs:minInclusive value="0"></xs:minInclusive>
                    <xs:maxInclusive value="9999"></xs:maxInclusive>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="text" type="xs:string" maxOccurs="1" minOccurs="1"></xs:element>
    </xs:sequence>
</xs:complexType>
</xs:schema>

```

1.1.5 Fehlerhandling

Treten während der Verarbeitung der Daten Fehler auf, wird eine veränderte Antwort generiert, die das folgende Format hat:

```
<?xml version="1.0" encoding="utf-8" ?>
<values xmlns="/InstantView/request/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="/InstantView/request/ file:powerProfile.xsd">
  <header>
    <name>Wirkleistung</name>
    <obis>OBIS 1-1:16.7.0</obis>
    <startts>130628130240</startts>
    <endts>130628130245</endts>
    <samplerate>1</samplerate>
    <no>0</no>
    <error>true</error>
  </header>
  <error>
    <id>[4-stellige Fehlernummer]</id>
    <text>[Fehlertext]</text>
  </error>
</values>
```

Folgende Fehlertexte können gesendet werden:

- 8100: "Value ts out of range" (Der Wert des Parameters ts wird nicht unterstützt, z.B. ts < 0)
- 8101: "Value n out of range" (Der Wert des Parameters n wird nicht unterstützt)
- 8102: "Unknown value for parameter resformat" (Der Wert des Parameters resformat wird nicht unterstützt)
- 8000: "Unknown parameter [Parameter-Name]" (Fehler, falls dem Aufruf ein nicht unterstützter Parameter mitgegeben wird)
- 8001: "Unknown parameter value" (Dem Parameter wurde ein Wert mitgegeben, der nicht verarbeitet werden kann)
- 8010: "Meter communication error" (Fehler bei der Kommunikation mit dem Zähler)
- 8011: "Profile not available" (Es ist kein Lastprofil verfügbar)
- 8012: "Resulting profile is oversized" (Das Ergebnis der Abfrage übersteigt die Pufferkapazität des Moduls)

Hinweis:

Das ComModul kann maximal 63kByte Dateien über die Schnittstelle zur Verfügung stellen. Wenn eine Anfrage ein Ergebnis liefert, das diese Größe überschreitet, wird die Fehler-ID „8012“ generiert.

Sollten einzelne Werte nicht auslesbar sein, wird keine Fehlermeldung generiert. Die entsprechenden Einträge erhalten die folgenden Werte:

- NAN falls es sich um einen Messwert <v> handelt
- ERR bei allen anderen Ergebniswerten – falls ein Fehler festgestellt wurde

- <leerer Eintrag> / Fehlender Tag: bei allen anderen Ergebniswerten, sonst - also falls der Wert bzw. Text nicht geliefert werden konnte und auch kein Fehler bei der Beschaffung des Werts/Texts festgestellt werden konnte.

1.2 Abfrage der ComModul-Uhrzeit (Veraltet – Deprecated)

Die ComModul-Uhrzeit wird mit der Uhrzeit des Zählers und der Atomzeit der PTB automatisch synchron gehalten und sollte beim Abfragen von ComModul-Daten als Referenzzeit für die abfragenden Systeme dienen. Zu diesem Zweck ist es möglich, mit der XML-Schnittstelle die ComModul-Uhrzeit abzufragen und zur Synchronisation von externen Systemen zu verwenden (Neu ab CM-Version 1.1.0, nicht in CM-Version v0.5.x).

Da sowohl die XML-Schnittstelle des ComModul als auch im Allgemeinen die verwendete Netzwerkverbindung nicht echtzeitfähig sind, kann die Übermittlung der „genauen“ Uhrzeit nicht garantiert werden. Bei einer hinreichend schnellen Netzwerkkumgebung kann von einer Latenzzeit der Antwortrückgabe von ca. 1 Sekunde ab Eingang beim ComModul-Web-Server ausgegangen werden. Genaueres ist hier jedoch noch ggf. experimentell zu bestimmen.

Die Abfrage der Uhrzeit erfolgt über einen HTTP-GET-Aufruf der Seite getDateTi-me.html und zugehörigen Parametern.

1.2.1 Format des GET-Requests

Die URL zur Abfrage der Uhrzeit lautet wie folgt – für fehlende Parameter werden jeweils die unten angegebenen Default-Werte verwendet:

`http://<IP des ComModul>/InstantView/request/getDateTime.html`

Mit den folgenden Parametern kann die Abfrage der im ComModul vorgehaltenen Daten eingegrenzt werden:

resformat	Format der Antwort (Veraltet – Deprecated)
param	(Veraltet – Deprecated)

Beispiel: Abfrage mit ausformulierten Default-Werten für die möglichen Parameter:

`http://<IP des ComModul>/InstantView/request/getDateTime.html`

Formatiert: Englisch (USA)

1.2.2 Format der Antwort

Das Format der Antwort sieht wie folgt aus (Angaben in eckigen Klammern stehen stellvertretend für den Typ bzw. Inhalt der dort zu erwartenden Angabe):

```
<?xml version="1.0" encoding="utf-8" ?>
<values xmlns="/InstantView/request/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="/InstantView/request/ file:dateTime.xsd">
  <header>
    <name>[Angefordertes Parameter]</name>
```

```

    <resformat>[Ergebnisformat]</resformat>
    <error>[true/false]</error>
  </header>
  <datetime>[Ergebnis-Datum und -Uhrzeit]</datetime>
</values>

```

Die Elemente der XML-Tags haben hierbei die folgende Bedeutung:

name	Name des Parameters (identisch zur Anfrage).
resformat	Ergebnisformat (identisch zum Parameter resformat in der Anfrage).
error	Falls ein Fehler aufgetreten ist „true“ - ansonsten „false“
datetime	Ergebniswert (Datum und Uhrzeit im Format wie unter <resformat> angegeben

Beispiel:

Es wird die aktuelle Uhrzeit im Standardformat (STD) geliefert. Die Uhrzeit zum Zeitpunkt der Anfrage war 13:02:41, Das Datum der 28.06.2013.

```

<?xml version="1.0" encoding="utf-8" ?>
<values xmlns="/InstantView/request/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="/InstantView/request/ file:dateTime.xsd">
  <header>
    <name>DTM</name>
    <resformat>STD</resformat>
    <error>>false</error>
  </header>
  <datetime>130628130241</datetime>
</values>

```

1.2.3 Formale Formatdefinition

Die formale Formatdefinition ist in der Datei /InstantView/request/dateTime.xsd hinterlegt:

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="/InstantView/request/"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:cm="/InstantView/request/"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation>
      Antwort des ComModuls auf getDateime-Requests
    </xs:documentation>
  </xs:annotation>
  <xs:element name="values" type="cm:t_values"></xs:element>

  <xs:complexType name="t_values">
    <xs:sequence>
      <xs:element name="header" type="cm:t_header" maxOccurs="1" minOccurs="1">
        </xs:element>
      <xs:choice>
        <xs:element name="datetime" minOccurs="1" maxOccurs="1">
          <xs:simpleType value="([0-9]{12}[nws]?) {0,1}">
            <xs:restriction base="xs:string">
              <xs:pattern value="([0-9]{12}[nws]?) {0,1}">
                </xs:pattern>
              <xs:pattern value="ERR"></xs:pattern>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>

```

```

        <xs:element name="error" type="cm:t_error" maxOccurs="1" minOc-
curs="1">
        </xs:element>
    </xs:choice>
</xs:sequence>
</xs:complexType>

<xs:complexType name="t_header">
    <xs:sequence>
        <xs:element name="name" maxOccurs="1" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:enumeration value="DTM"></xs:enumeration>
                    <xs:enumeration value="ERR"></xs:enumeration>
                    <xs:enumeration value=""></xs:enumeration>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="resformat" maxOccurs="1" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:enumeration value="STD"></xs:enumeration>
                    <xs:enumeration value="ERR"></xs:enumeration>
                    <xs:enumeration value=""></xs:enumeration>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="error" type="xs:boolean" maxOccurs="1" minOc-
curs="1">
        </xs:element>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="t_error">
    <xs:sequence>
        <xs:element name="id" maxOccurs="1" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:positiveInteger">
                    <xs:minInclusive value="0"></xs:minInclusive>
                    <xs:maxInclusive value="9999"></xs:maxInclusive>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="text" type="xs:string" maxOccurs="1" minOccurs="1">
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:schema>

```

1.2.4 Fehlerhandling

Treten während der Verarbeitung der Daten Fehler auf, wird eine veränderte Antwort generiert, die das folgende Format hat:

```
<?xml version="1.0" encoding="utf-8" ?>
<values xmlns="/InstantView/request/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="/InstantView/request/ file:dateTime.xsd">
  <header>
    <name>[Angeforderter Parameter]</name>
    <resformat>[Ergebnisformat]</resformat>
    <error>true</error>
  </header>
  <error>
    <id>[4-stellige Fehlernummer]</id>
    <text>[Fehlertext]</text>
  </error>
</values>
```

Formatiert: Englisch (USA)

Folgende Fehlertexte können gesendet werden:

- 8102: "Unknown value for parameter resformat" (Der Wert des Parameters resformat wird nicht unterstützt)
- 8000: "Unknown parameter [Parameter-Name]" (Fehler, falls dem Aufruf ein nicht unterstützter Parameter mitgegeben wird)
- 8001: "Unknown parameter value" (Der Wert des mitgegebenen Parameter kann nicht verarbeitet werden)
- 8010: "meter communication error" (Fehler bei der Kommunikation mit dem Zähler)

Sollten einzelne Werte nicht auslesbar sein, wird keine Fehlermeldung generiert. Die entsprechenden Einträge erhalten die folgenden Werte:

- ERR bei allen Header-Werten – falls ein Fehler festgestellt wurde
- <leerer Eintrag> / Fehlender Tag: bei allen anderen Ergebniswerten, auch Nicht-Header-Werten - also falls der Wert bzw. Text nicht geliefert werden konnte und auch kein Fehler bei der Beschaffung des Werts/Texts festgestellt werden konnte.

1.3 Abfrage von Identifikationsmerkmalen und Systeminformationen (Veraltet – Deprecated)

Zur Identifikation des ComModul und zur Abfrage von Systeminformationen sind folgende Parameter abrufbar:

1. UPnP-Name
2. Zählernummer
3. FA-Nummer
4. Seriennummer CPU (Hardware-Nummer)
5. MAC-ID
6. Hardware-Version
7. Firmware-Version (Software-Version Normalsystem)
8. System-IDs im Format <System 1>:<System2>... (Versionsbezeichnungen der Systemkomponenten, siehe Spezifikation zu CMS1019)
9. Lieferanten-ID
10. Letzter Systemstart
11. Anzahl Bootvorgänge

Die Abfrage der Identifikationsmerkmale erfolgt über einen HTTP-GET-Aufruf der Seite `getIdentification.html` und zugehörigen Parametern.

1.3.1 Format des GET-Requests

Die URL zur Abfrage der Uhrzeit lautet wie folgt – für fehlende Parameter werden jeweils die unten angegebenen Default-Werte verwendet:

```
http://<IP des ComModul>/InstantView/request/getIdentification.html
```

Mit den folgenden Parametern kann die Abfrage der im ComModul vorgehaltenen Daten eingegrenzt werden:

resformat	Format der Antwort (Default: "STD"). STD: In der Antwort werden alle Identifikationsmerkmale aufgelistet.
-----------	--

Beispiel:

Abfrage mit ausformulierten Default-Werten für die möglichen Parameter:

```
http://<IP des ComModul>/InstantView/request/getIdentification.html?  
resformat=STD
```

1.3.2 Format der Antwort

Das Format der Antwort sieht wie folgt aus (Angaben in eckigen Klammern stehen stellvertretend für den Typ bzw. Inhalt der dort zu erwartenden Angabe):

```
<?xml version="1.0" encoding="utf-8" ?>
```

```
<system xmlns="/InstantView/request/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="/InstantView/request/ file:identification.xsd">
  <header>
    <resformat>[Ergebnisformat]</resformat>
    <error>[true/false]</error>
  </header>
  <id>
    <upnp>...</upnp>
    <meter>...</meter>
    <fa>...</fa>
    <cpu>...</cpu>
    <mac>...</mac>
  </id>
  <version>
    <hw>...</hw>
    <sw>...</sw>
    <sys>...</sys>
  </version>
  <supplier>
    <supplierid>...</supplierid>
  </supplier>
  <watchdog>
    <laststart>...</laststart>
    <bootcount>...</bootcount>
  </watchdog>
</system>
```

Formatiert: Englisch (Großbritannien)

Innerhalb des XML-Blocks <system> sind die Angaben Unterblöcken zugeordnet:

header	Kopf mit allgemeinen Informationen zur gesamten Nachricht
id	Block mit Identifikationsparametern zum ComModul
version	Block mit Versionsinformationen zum ComModul
supplier	Block mit Informationen bzgl. des eingestellten Lieferanten
watchdog	Block mit Informationen zum Betrieb des ComModul

Die Elemente der XML-Tags in den einzelnen Blöcken haben hierbei die folgende Bedeutung:

resformat	Ergebnisformat (identisch zum Parameter resformat in der Anfrage).
error	Falls ein Fehler aufgetreten ist „true“ - ansonsten „false“
upnp	UPnP-Name
meter	Zählernummer
fa	FA-Nummer
cpu	Serialnummer CPU (Hardware-Nummer)
mac	MAC-ID
hw	Hardware-Version
sw	Firmware-Version (Software-Version Normalsystem)
sys	System-IDs im Format <System 1>:<System2>... (also die Versionsbezeichnungen der Systemkomponenten, siehe Spezifikation zu CMS1019)
supplierid	Lieferanten-ID
laststart	Letzter Systemstart
bootcount	Anzahl Bootvorgänge

Beispiel:

Es werden die Identifikationsmerkmale im Standardformat geliefert:

```
<?xml version="1.0" encoding="utf-8" ?>
<system xmlns="/InstantView/request/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="/InstantView/request/ file:identification.xsd">
  <header>
    <resformat>STD</resformat>
    <error>>false</error>
  </header>
  <id>
    <upnp>Mein Zaehler</upnp>
    <meter>11664711</meter>
    <fa>30008733</fa>
    <cpu>00158</cpu>
    <mac>00:03:12:01:00:9E</mac>
  </id>
  <version>
    <hw>001</hw>
    <sw>WIKI011000</sw>
    <sys>FSUB011000:FSUD011000</sys>
  </version>
  <supplier>
    <supplierid>enbw</supplierid>
  </supplier>
  <watchdog>
    <laststart>130907141334</laststart>
    <bootcount>8</bootcount>
  </watchdog>
</system>
```

Formatiert: Englisch (Großbritannien)

Formatiert: Englisch (Großbritannien)

1.3.3 Formale Formatdefinition

Die formale Formatdefinition ist in der Datei /InstantView/request/identification.xsd hinterlegt:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="/InstantView/request"
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:cm="/InstantView/request" element-
FormDefault="qualified" attributeFormDefault="unqualified">
  <xs:annotation>
    <xs:documentation>
      Antwort des ComModuls auf getIdentification-Requests
    </xs:documentation>
  </xs:annotation>
  <xs:element name="system" type="cm:t_system"></xs:element>

  <xs:complexType name="t_system">
    <xs:sequence>
      <xs:element name="header" type="cm:t_header" maxOccurs="1" minOccurs="1">
        </xs:element>
      <xs:choice>
        <xs:group ref="cm:SystemData"></xs:group>
        <xs:element name="error" type="cm:t_error" maxOccurs="1"
minOccurs="1">
          </xs:element>
        </xs:choice>
      </xs:sequence>
    </xs:complexType>

  <xs:complexType name="t_header">
    <xs:sequence>
      <xs:element name="resformat" maxOccurs="1" minOccurs="1">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="STD"></xs:enumeration>
            <xs:enumeration value="ERR"></xs:enumeration>
            <xs:enumeration value=""></xs:enumeration>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
```



```

        </xs:simpleType>
    </xs:element>
    <xs:element name="error" type="xs:boolean" maxOccurs="1" minOccurs="1">
    </xs:element>
</xs:sequence>
</xs:complexType>

<xs:complexType name="t_id">
    <xs:sequence>
        <xs:element name="upnp" maxOccurs="1" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:pattern value=".*"/>
                    <xs:pattern value="ERR"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="meter" maxOccurs="1" minOccurs="1">
            <xs:simpleType value="[0-9]*">
                <xs:restriction base="xs:positiveInteger">
                    <xs:pattern value="[0-9]*"/>
                    <xs:pattern value="ERR"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="fa" maxOccurs="1" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:positiveInteger">
                    <xs:pattern value="[0-9]*"/>
                    <xs:pattern value="ERR"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="cpu" maxOccurs="1" minOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:positiveInteger">
                    <xs:pattern value="[0-9]*"/>
                    <xs:pattern value="ERR"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="mac" maxOccurs="1" minOccurs="1">
            <xs:simpleType value="([0-9a-fA-F]{2}){5}[0-9a-fA-F]{2}){0,1}">
                <xs:restriction base="xs:string">
                    <xs:pattern value="([0-9a-fA-F]{2}){5}[0-9a-fA-F]{2}){0,1}">
                    </xs:pattern>
                    <xs:pattern value="ERR"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="t_version">
    <xs:sequence>
        <xs:element name="hw" maxOccurs="1" minOccurs="1">
            <xs:simpleType value="([0-9A-Z]{3,5}){0,1}">
                <xs:restriction base="xs:string">
                    <xs:pattern value="([0-9A-Z]{3,5}){0,1}">
                    <xs:pattern value="ERR"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="sw" maxOccurs="1" minOccurs="1">
            <xs:simpleType value="(WIKI[0-9]{6}){0,1}">
                <xs:restriction base="xs:string">
                    <xs:pattern value="(WIKI[0-9]{6}){0,1}">
                    <xs:pattern value="ERR"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <xs:element name="sy" maxOccurs="1" minOccurs="1">
            <xs:simpleType value="([A-Z]{4}[0-9]{6})??">
                <xs:restriction base="xs:string">
                    <xs:pattern value="([A-Z]{4}[0-9]{6})??">
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
    </xs:sequence>
</xs:complexType>

```

Formatiert: Englisch (USA)

```

        <xs:pattern value="ERR"></xs:pattern>
    </xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>

<xs:complexType name="t_supplier">
<xs:sequence>
<xs:element name="supplierid" maxOccurs="1" minOccurs="1">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="enbw"></xs:enumeration>
<xs:enumeration value="yello"></xs:enumeration>
<xs:enumeration value="watt"></xs:enumeration>
<xs:enumeration value="natur"></xs:enumeration>
<xs:enumeration value="ERR"></xs:enumeration>
<xs:enumeration value=""></xs:enumeration>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>

<xs:complexType name="t_watchdog">
<xs:sequence>
<xs:element name="laststart" maxOccurs="1" minOccurs="1">
<xs:simpleType value="([0-9]{12}[nws]?) {0,1}">
<xs:restriction base="xs:string">
<xs:pattern value="([0-9]{12}[nws]?) {0,1}"></xs:pattern>
<xs:pattern value="ERR"></xs:pattern>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="bootcount" maxOccurs="1" minOccurs="1">
<xs:simpleType>
<xs:restriction base="xs:positiveInteger">
<xs:pattern value="[0-9]*"></xs:pattern>
<xs:pattern value="ERR"></xs:pattern>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>

<xs:group name="SystemData">
<xs:sequence><xs:element name="id" type="cm:t_id" maxOccurs="1" minOccurs="1">
</xs:element><xs:element name="version" type="cm:t_version" maxOccurs="1" minOccurs="1">
</xs:element><xs:element name="supplier" type="cm:t_supplier" maxOccurs="1" minOccurs="1">
</xs:element><xs:element name="watchdog" type="cm:t_watchdog" maxOccurs="1" minOccurs="1">
</xs:element></xs:sequence>
</xs:group>

<xs:complexType name="t_error">
<xs:sequence>
<xs:element name="id" maxOccurs="1" minOccurs="1">
<xs:simpleType>
<xs:restriction base="xs:positiveInteger">
<xs:minInclusive value="0"></xs:minInclusive>
<xs:maxInclusive value="9999"></xs:maxInclusive>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="text" type="xs:string" maxOccurs="1" minOccurs="1"></xs:element>
</xs:sequence>
</xs:complexType>
</xs:schema>

```

1.3.4 Fehlerhandling

Treten während der Verarbeitung der Daten Fehler auf, wird eine veränderte Antwort generiert, die das folgende Format hat:

```
<?xml version="1.0" encoding="utf-8" ?>
<system xmlns="/InstantView/request/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="/InstantView/request/ file:identification.xsd">
  <header>
    <resformat>[Ergebnisformat]"</resformat>
    <error>true</error>
  </header>
  <error>
    <id>[4-stellige-Fehlernummer]</id>
    <text>[Fehlertext]</text>
  </error>
</system>
```

Folgende Fehlertexte können gesendet werden:

- 8102: "Unknown value for parameter resformat" (Der Wert des Parameters resformat wird nicht unterstützt)
- 8000: "Unknown parameter [Parameter-Name]" (Fehler, falls dem Aufruf ein nicht unterstützter Parameter mitgegeben wird)
- 8001: "Unknown parameter value" (Unbekannter Parameterwert für sonstigen Parameter)
- 8010: "Meter communication error" (Fehler bei der Kommunikation mit dem Zähler)

Sollten einzelne Werte nicht auslesbar sein, wird keine Fehlermeldung generiert. Die entsprechenden Einträge erhalten die folgenden Werte:

- ERR bei allen Header-Werten – falls ein Fehler festgestellt wurde
- <leerer Eintrag> / Fehlender Tag: bei allen anderen Ergebniswerten, auch Nicht-Header-Werten - also falls der Wert bzw. Text nicht geliefert werden konnte und auch kein Fehler bei der Beschaffung des Werts bzw. Texts festgestellt werden konnte.